CE216623 – FM0+ I2C Master S6E1B

This application demonstrates I2C communication between a master (this project) and slave (CE216624_I2C_Slave_S6E1B). The master periodically sends an on/off command to the slave, which controls an LED accordingly.

Overview

This Code Example requires two boards in order to show I2C communication. The master connects to the slave and periodically sends single-byte commands. The slave (CE216624_I2C_Slave_S6E1B) reads these commands and turns its LED on or off as commanded.

Requirements

Tool: PSoC Creator 4.0 and Peripheral Driver Library (PDL) 2.1

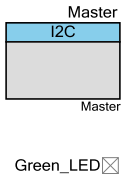
Programming Language: C (GCC 4.9.3)

Associated Parts: All S6E1B parts

Related Hardware: FM0-100L-S6E1B8

Design

The schematic file includes the MFS (I2C) and a GPIO Component, renamed as shown below.



The firmware performs following functions:

1. Initialize the master LED (off)
2. Select the I2C pins and enable pull-up resistors
3. Initialize the MFS block for I2C
4. Send the slave address
5. Start a transmit communication
6. Send the one-byte command
7. Complete the transmit communication
8. Toggle the master LED
9. Change the command
10. Send another command (step #4)

Design Considerations

PDL Installation

The project assumes that you have installed the PDL in the location specified in the **Project Management** panel of the **Tools** > **Options** dialog. If that location is incorrect you will see the build error “The given PDL path is invalid. Unable to find required PDSC file.” To correct this problem in a newly-created project, open the **Project** > **Properties** dialog and enter the correct path to the PDL. To avoid the problem in projects you create in the future, make sure you put the correct path in the **Tools** > **Options** dialog.

Hardware Setup

The LED GPIO is connected to a green LED.

The I2C pins are connected to header CN7 on the kit. SCK is pin 10 (SCK1 on the silk screen) and SDA is pin 9 (SDA1 on the silk screen). It is good practice to also connect GND between the kits using pin 7 on CN7 (or another convenient GND pin).

Table 1 lists the pin connections required to use this code example on FM0-100L-S6E1B8 kits.

Table 1. List of Pins

Pin	FM0-100L-S6E1B8
Green_LED:GPIO	P3E
Master:SCK	P4D
Master:SOT	P4C

Components

Table 2 lists the PSoC Creator Components used in this example, as well as the hardware resources used by each.

Table 2. List of PSoC Creator Components

Component	Version	Hardware Resources
PDL_MFS	1.0	MFS block in I2C mode
PDL_GPIO	1.0	GPIO pin

Parameter Settings

The “Multi-Function Serial I2C” component macro uses mostly default parameter settings, with the following modifications.

Table 3: Component Settings

Tab	Setting	Value
None	Name	Master
Basic	MFSConfig	I2C
I2C	enI2cMsMode	Master
FIFO	u8ByteCount1	0
	u8ByteCount2	0

Operation

Make the wiring connections between two kits as described in Hardware Setup above (power should be detached from the kits).

Connect the kits to your PC using the USB cables. Program one kit with the master (this project) and another with the slave (CE216624_I2C_Slave_S6E1B) project. The application blinks the green LED on both kits in unison.

If the application does not work immediately press and hold the reset buttons on both kits, then release the slave kit, followed by the master. This ensures that the slave is “listening” when the master initiates communication.

Note that it is also possible to use this master with slave projects on the FM0-V48-S6E1A1 and FM0-64L-S6E1C3 kits. You must carefully wire the SDA and SCL pins according to the documentation for each kit.

Related Documents

Table 4 lists relevant application notes, code examples, knowledge base articles, device datasheets, and Component datasheets.

Table 4. Related Documents

PSoC Creator Component Datasheets	
PDL_MFS	Supports I2C, UART, LIN and CSIO (SPI) modes (right-click on the Component to access)
PDL_GPIO	Supports firmware access to physical pins (right-click on the Component to access)
Device Documentation	
S6E1B	FM0+ S6E1B-Series Ultra Low Power ARM® Cortex®-M0+ Microcontroller (MCU) Family
Development Kit (DVK) Documentation	
FM0-100L-S6E1B8	ARM® Cortex®-M0+ MCU Starter Kit with USB and SD Card Interface

Document History

Document Title: CE216623 – FM0+ I2C Master S6E1B

Document Number: 002-16623

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	5436964	YFS	09/14/16	New Code Example.
*A	5453421	YFS	02/28/16	Changed the workspace folder name. Renamed the PDF file. Moved the PDF file. Corrected the Documentation entry in XML file.
*B	5776609	YFS	6/16/17	Added search keyword so that user can quickly find Code Examples from the component instance popup menu. Updated logo and copyright date.

Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at Cypress Locations.

Products

ARM® Cortex® Microcontrollers	cypress.com/arm
Automotive	cypress.com/automotive
Clocks & Buffers	cypress.com/clocks
Interface	cypress.com/interface
Lighting & Power Control	cypress.com/powerpsoc
Memory	cypress.com/memory
PSoC	cypress.com/psoc
Touch Sensing	cypress.com/touch
USB Controllers	cypress.com/usb
Wireless/RF	cypress.com/wireless

PSoC® Solutions

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP

Cypress Developer Community

Forums | Projects | Videos | Blogs | Training | Components

Technical Support

cypress.com/support

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709

Phone : 408-943-2600
Fax : 408-943-4730
Website : www.cypress.com

© Cypress Semiconductor Corporation, 2016-2017. This document is the property of Cypress Semiconductor Corporation and its subsidiaries, including Spansion LLC ("Cypress"). This document, including any software or firmware included or referenced in this document ("Software"), is owned by Cypress under the intellectual property laws and treaties of the United States and other countries worldwide. Cypress reserves all rights under such laws and treaties and does not, except as specifically stated in this paragraph, grant any license under its patents, copyrights, trademarks, or other intellectual property rights. If the Software is not accompanied by a license agreement and you do not otherwise have a written agreement with Cypress governing the use of the Software, then Cypress hereby grants you a personal, non-exclusive, nontransferable license (without the right to sublicense) (1) under its copyright rights in the Software (a) for Software provided in source code form, to modify and reproduce the Software solely for use with Cypress hardware products, only internally within your organization, and (b) to distribute the Software in binary code form externally to end users (either directly or indirectly through resellers and distributors), solely for use on Cypress hardware product units, and (2) under those claims of Cypress's patents that are infringed by the Software (as provided by Cypress, unmodified) to make, use, distribute, and import the Software solely for use with Cypress hardware products. Any other use, reproduction, modification, translation, or compilation of the Software is prohibited.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS DOCUMENT OR ANY SOFTWARE OR ACCOMPANYING HARDWARE, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. To the extent permitted by applicable law, Cypress reserves the right to make changes to this document without further notice. Cypress does not assume any liability arising out of the application or use of any product or circuit described in this document. Any information provided in this document, including any sample design information or programming code, is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. Cypress products are not designed, intended, or authorized for use as critical components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or system could cause personal injury, death, or property damage ("Unintended Uses"). A critical component is any component of a device or system whose failure to perform can be reasonably expected to cause the failure of the device or system, or to affect its safety or effectiveness. Cypress is not liable, in whole or in part, and you shall and hereby do release Cypress from any claim, damage, or other liability arising from or related to all Unintended Uses of Cypress products. You shall indemnify and hold Cypress harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of Cypress products.

Cypress, the Cypress logo, Spansion, the Spansion logo, and combinations thereof, WICED, PSoC, CapSense, EZ-USB, F-RAM, and Traveo are trademarks or registered trademarks of Cypress in the United States and other countries. For a more complete list of Cypress trademarks, visit cypress.com. Other names and brands may be claimed as property of their respective owners.